

日 本 国 特 許 庁  
JAPAN PATENT OFFICE

別紙添付の書類に記載されている事項は下記の出願書類に記載されている事項と同一であることを証明する。

This is to certify that the annexed is a true copy of the following application as filed with this Office

出 願 年 月 日

Date of Application:

2000年12月19日

出 願 番 号

Application Number:

特願2000-389956

出 願 人

Applicant(s):

株式会社日立製作所

株式会社 日立システムアンドサービス

USSN 10/015,800

MATTINGLY, STANGER + MALUR

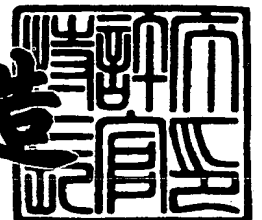
(703) 684-1120

DKT: ASA-1046

2001年12月21日

特 許 庁 長 官  
Commissioner,  
Japan Patent Office

及 川 耕 造



出証番号 出証特2001-3109675

【書類名】 特許願

【整理番号】 K00020271

【提出日】 平成12年12月19日

【あて先】 特許庁長官殿

【国際特許分類】 G06F 17/30

【請求項の数】 10

【発明者】

    【住所又は居所】 神奈川県川崎市幸区鹿島田 8 9 0 番地 株式会社日立製作所 ビジネスソリューション開発本部内

    【氏名】 多田 勝己

【発明者】

    【住所又は居所】 東京都大田区大森北三丁目 2 番 1 6 号 株式会社日立システムアンドサービス内

    【氏名】 小泉 直弘

【発明者】

    【住所又は居所】 東京都大田区大森北三丁目 2 番 1 6 号 株式会社日立システムアンドサービス内

    【氏名】 高取 壽

【特許出願人】

    【識別番号】 000005108

    【氏名又は名称】 株式会社日立製作所

【特許出願人】

    【識別番号】 391002409

    【氏名又は名称】 株式会社日立システムアンドサービス

【代理人】

    【識別番号】 100075096

    【弁理士】

    【氏名又は名称】 作田 康夫

【手数料の表示】

【予納台帳番号】 013088

【納付金額】 21,000円

【提出物件の目録】

【物件名】 明細書 1

【物件名】 図面 1

【物件名】 要約書 1

【プルーフの要否】 要

【書類名】 明細書

【発明の名称】 文書情報の検索方法

【特許請求の範囲】

【請求項 1】

イメージ文書を対象とした文字認識処理を実行した結果出力されるテキストによる文書を対象として、検索者が指定した検索文字列を含む文書を検索するシステムにおいて、前記検索文字列を所定の  $n$  文字単位の部分文字列 ( $n \geq 2$ ) に分割する検索用文字列分割ステップと、前記  $n$  文字単位の部分文字列 ( $n \geq 2$ ) に対して、誤認識される可能性の高い文字形状の類似した類似文字列を格納することにより予め作成した  $n$  文字単位の類似文字テーブルを参照する類似文字テーブル参照ステップと、前記検索文字列を構成する部分文字列に対して  $n$  文字単位類似文字テーブルを参照することにより抽出し類似文字列群を組合せて展開語を生成する検索文字列展開ステップを有することを特徴とする検索文字列の展開方法。

【請求項 2】

請求項 1 記載の検索文字列の展開方法において、字単位類似文字テーブルの見出し文字は、 $n$  文字の組合せにより構成される部分文字列群のうちの一部の組合せのみを格納したことを特徴とする検索文字列の展開方法。

【請求項 3】

請求項 2 記載の検索文字列の展開方法において、前記検索タームを構成する部分文字列が前記  $n$  文字単位類似文字テーブル中に存在しなかった場合には、該当の部分文字列に対して類似文字列の抽出を行わないことを特徴とする検索文字列の展開方法。

【請求項 4】

請求項 2 記載の検索文字列の展開方法において、前記検索タームを構成する部分文字列が前記  $n$  文字単位類似文字テーブル中に存在しなかった場合には、予め  $m$  文字単位 ( $m < n$ ) について誤認識される可能性の高い文字形状の類似した類似文字を格納した  $m$  文字単位類似文字テーブルを参照して、展開語を生成することを特徴とする検索文字列の展開方法。

【請求項 5】

請求項 1 記載の検索文字列の展開方法において、前記検索文字列に対して文字列長を算出し、前記検索文字列長に応じて展開語の生成方法を切り替える展開方法切り替えステップを有することを特徴とする検索文字列の展開方法。

【請求項 6】

イメージ文書を対象とした文字認識処理を実行した結果出力されるテキストによる文書を対象として、検索者が指定した検索文字列を含む文書を検索するシステムにおいて、前記検索文字列に対して検索文字列長を算出し、前記検索文字列長に応じて展開の方法を切り替える展開方法切り替えるステップを有することを特徴とする検索文字列の展開方法。

【請求項 7】

請求項 5 記載の検索文字列の展開方法において、前記検索文字列長に応じて生成する前記展開文字列の数を調整することを特徴とする検索文字列の展開方法。

【請求項 8】

請求項 6 記載の検索文字列の展開方法において、前記ターム長に応じて展開語を生成する、しないを選択することを特徴とする検索文字列の展開方法。

【請求項 9】

請求項 8 記載の検索文字列の展開方法において、前記展開方法を切り替えるための設定情報を有することを特徴とする検索文字列の展開方法。

【請求項 10】

請求項 9 記載の検索文字列の展開方法において、得られた検索文字列を、それらの論理和の条件として検索を実行するテキストサーチステップを有することを特徴とする文書情報の検索方法。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】

本発明は、紙の文書を電子化して保管・管理するシステムにおける文書情報の検索方法に関するものである。

## 【0002】

## 【従来の技術】

情報化社会の本格的な進展に伴い、文書を紙のままファイルして保管・管理していた従来の文書管理方法に変わり、文書を電子化して保管・管理する文書管理システムによる管理方法が普及してきた。当初は、紙の形態の文書をスキャナで取り込みイメージデータを生成し、そのイメージデータに対して「作成者」や「日付」、「キーワード」等の書誌情報を関連づけて登録し、検索のときには書誌情報を検索の対象とすることで所望する文書を検索していた。しかし、書誌情報を用いた検索だけでは所望する文書を見つけ出すことが困難であり、また本文テキストを対象とする全文検索技術が実用化されていることから、イメージ文書の世界にも全文検索を行う機能を有する文書管理システムが普及してきた。

## 【0003】

この文書管理システムでは、まず紙の形態の文書をスキャナで取り込みイメージデータとして蓄積し、更にイメージデータから文字認識して得られたテキストデータを併せて蓄積しておく。そして、検索時にはテキストデータを対象とした全文検索を行い、結果表示の際には指定したテキストデータまたは、それに対応するイメージデータを表示するシステムである。全文検索では基本的に誤りがないテキストデータを対象とするのが前提であるが、検索対象のテキストデータはイメージデータからOCR（光学的文字認識装置）による文字認識により生成されているため、認識誤りを含んだテキストデータとなる可能性がある。そのため、正しく文字認識されていれば本来ヒットすべきテキストデータが、認識誤りを含むことでヒットせずに検索漏れとなる場合がある。

## 【0004】

この検索漏れの発生の問題を解決するために、従来からOCRによる認識誤りのあるテキストデータは人手により校正を行っていた。すなわち、文書登録の際にOCR出力のテキストデータに対して、元の文書と比較して誤っている箇所を見つけだし人手によりテキストデータを修正することで登録文書の認識誤りを解消し、文書検索を可能とするものである。しかし、上記の方法では人手による修正作業であるために、ユーザの負担が大きく文書登録に要する手間と時間がかか

るという問題があった。

【0005】

この問題点を解決する技術として特開平4-158478号公報では、検索対象の曖昧さをある程度許容して検索を行う方法が開示されている。上記従来技術では、文書登録の時にOCR出力によるテキストデータには修正を加えず、そのまま文書登録を行う。つまり、誤りを含んだテキストのまま文書登録を行うことで人手による修正作業を必要とせずに検索時に工夫がなされていることに特徴がある。

【0006】

以下、従来技術の認識誤りを許容した検索方法について、図2に示すフローチャートに従い説明する。この方法では、OCRが認識誤りする可能性のある文字形状の類似した候補文字（以下、候補文字とする）を予め1文字単位の類似文字テーブルに列挙しておく。まず、検索者は探したい文書に含まれている検索文字列を入力する（ステップ1000）。次に入力された検索文字列を1文字毎に分割し（ステップ1001）、分割された各文字に対して上記1文字単位の類似文字テーブルから候補文字を参照し（ステップ1002）、参照した各文字の候補文字を組合せて複数の文字列（以下、展開語とする）に展開する（ステップ1003）。次にそれらの展開語のいずれかと一致する文書を探すために展開語の論理和（OR）集合による全文検索を行い（ステップ1004）、その検索結果を取得する（ステップ1005）。このように認識誤りする可能性のある展開語も含めて検索を行うことで、OCRによる認識誤りが生じても検索漏れとならずに検索を可能とするものである。

【0007】

【発明が解決しようとする課題】

しかしながら、上記従来技術では以下に示す問題がある。

【0008】

文書検索時に指定した検索文字列が長い場合は、認識誤りする可能性のある展開語の数が爆発的に増加し、それに伴い検索に要する時間が長くなるということである。

## 【0009】

例えば、検索文字列が“日本文化”の4文字の場合、各文字の候補文字が（日、目、白、日、白）（本、木、不、天、末）（文、丈、女、文、大）（化、仕、牝、比、北）のように各々5つずつと仮定すると、生成される展開語はすべての文字を組合せることで、 $5 \times 5 \times 5 \times 5 = 5^4 = 625$ 通りとなる。

## 【0010】

同様に、検索文字列が“lock”の4文字の場合、各文字の候補文字が（l、I、!、1、i）（o、O、0、Q、6）（c、C、G、e、q）（k、K、h、b、R）のように各々5つずつと仮定すると、生成される展開語はすべての文字を組合せることで、 $5 \times 5 \times 5 \times 5 = 5^4 = 625$ 通りとなる。

## 【0011】

さらに検索文字列長が長くなり、8文字の場合では展開語数が $5^8 = 390$ 、625通りにもなり、検索文字列が長くなるにつれて展開語数が大きく増加することは明らかである。そして、検索処理では展開語の論理和（OR）集合による全文検索のため、展開語数の増加は検索時間の増加となる。そのため、検索文字列が長くなるに伴い検索に要する時間が長大化する。

## 【0012】

上記従来の技術の問題点に対して、本発明の解決しようとする課題は検索文字列が長くても、検索漏れの発生を低減しつつ実用的な検索時間での検索を可能とするOCR認識誤りを許容した文書情報の検索方法を実現することである。

## 【0013】

## 【課題を解決するための手段】

上記の課題を、本発明では以下の処理から構成される文書情報の検索方法により解決する。図3のフローチャートに従い説明する。

## 【0014】

本発明による文書検索方法では、 $n$ 文字単位（ $n \geq 2$ ）の部分文字列に対しOCRが認識誤りする可能性のある候補文字を、予め $n$ 文字単位類似文字テーブルに列挙しておく。

## 【0015】

まず、検索者は探したい文書に含まれている検索文字列を入力する（ステップ1500）。入力された検索文字列を所定の $n$ 文字単位（ $n \geq 2$ ）の部分文字列に分割する検索用文字列分割ステップ（ステップ1501）と、分割された各部分文字列に対し前記 $n$ 文字単位類似文字テーブルを参照し、検索精度向上に寄与する可能性の高い候補文字列を抽出する $n$ 文字単位類似文字テーブル参照ステップ（ステップ1502）と、参照した各部分文字列の候補文字を組合せて展開語を生成する検索文字列展開ステップ（ステップ1503）と、その展開語のいずれかを含む文書を探すための検索条件式を生成する検索条件式生成ステップ（ステップ1504）、検索条件式をテキストサーチプログラムへ入力する検索条件入力ステップ（ステップ1505）からなる検索方法により実現する。

## 【0016】

ここで、上記類似文字テーブルの候補文字を $n$ 文字単位とすることにより、「1文字単位で誤認識される可能性が低い候補文字を組合せた文字列は、検索精度向上の観点で寄与する確率が低い」という特徴を利用し、精度向上に寄与しない $n$ 文字単位の候補文字を排除し候補数を削減している。

## 【0017】

## 【発明の実施の形態】

以下、本発明を適用した第一の実施例について図面を用いて説明する。まず、本発明を適用した文書検索システムの構成図を図1に示す。この文書検索システムは、ディスプレイ100、キーボード101、中央演算装置CPU102、スキャナ103、主メモリ200、磁気ディスク104から構成される。また、これらはバス105で接続されている。磁気ディスク104にはテキストデータ106、イメージデータ107、後述する各種プログラム108、類似文字テーブル109が格納される。

## 【0018】

主メモリ200には、システム制御プログラム201、文書登録制御プログラム202、スキャナ制御プログラム203、OCR制御プログラム204、文書登録プログラム205、展開制御プログラム206、展開語生成プログラム20

7、検索条件式生成プログラム211、検索制御プログラム212、検索条件式解析プログラム213、テキストサーチプログラム214、表示プログラム215が磁気ディスク104から読み出されて格納されるとともにワークエリア216が確保される。

#### 【0019】

展開語生成プログラム207は検索用文字列分割プログラム208、類似文字テーブル参照プログラム209、検索文字列展開プログラム210から構成されている。これらのプログラムはユーザのキーボード101からの指示に応じてシステム制御プログラム201の制御の下で実行される。以上、本文書管理システムの構成である。

#### 【0020】

次に類似文字テーブル109について説明する。一般的には $n$ 文字単位の候補文字を列挙しているが、本実施例では $n = 2$ の場合を例にして説明する。

#### 【0021】

通常OCRは文字の形の特徴に基づいて文字認識するので、常に確実な認識結果が得られるわけではなく確定文字の他にある程度可能性のある候補文字を用意している。本類似文字テーブルは、展開に漏れのないように全文字コードを組合せた2文字の学習データを用いてOCRが出力する候補文字を認識誤りする確率情報（以下、出現確率とする）と共に収集することで実現する。類似文字テーブル作成の概要を図4に示す。まず全文字コードを組合せた2文字の学習データを印字した紙文書をスキャナへ入力し、イメージデータを出力する（ステップ1550）。次にイメージデータをOCRへ入力し、候補文字とその出現確率を列挙した認識テキストデータを出力する（ステップ1551）。次に学習データの元テキストデータと前述の認識テキストデータを類似文字テーブル作成プログラムに入力し類似文字テーブルを作成する（ステップ1552）。

#### 【0022】

次に類似文字テーブル作成プログラムの詳細な処理手順を図5のフローチャートに示す。まず学習データの元テキストデータを入力（ステップ1600）し、見出し文字を1行ずつ読み出し類似文字テーブルに追加する（ステップ1601

）。次に認識テキストデータを入力（ステップ1602）し、各候補文字を出現確率と共に1行ずつ抽出する（ステップ1603）。次に各候補文字の出現確率が所定値を超える候補文字を類似文字テーブルに追加（ステップ1604～1606）していくことで類似文字テーブルを生成する。このとき、出現確率が所定の値を超える候補文字のみを抽出し類似文字テーブルに列挙することで検索精度の向上に寄与しない候補文字を排除し、大幅に候補文字を削減している。なお、上記の例における確率情報はOCR出力による出現確率を用いているが、同様の学習データに対して文字認識を複数回繰返して得られる学習結果による頻度情報であっても構わない。図6は類似文字テーブルの一例であり、縦の列が見出し文字（500）であり、横に見出し文字に対する認識誤りする可能性のある候補文字（501）を列挙している。

#### 【0023】

次に候補文字の単位をn文字とする効果について“日本”の場合を例にして説明する。従来技術では各文字の候補文字が認識誤りし易い順に列挙された（日、目、白、曰、臼）（本、木、不、天、末）に対して、すべてを組合せた

（日本、目本、白本、曰本、臼本、  
 日木、目木、白木、曰木、臼木、  
 日不、目不、白不、曰不、臼不、  
 日天、目天、白天、曰天、臼天、  
 日末、目末、白末、曰末、臼末）

の $5 \times 5 = 25$ 通りの展開語を生成し、それらの展開語の論理和集合を検索条件として検索を行っている。しかし、“臼末”のような第5候補文字と第5候補文字の組合せの展開語が検索精度の向上に寄与する可能性は極めて低いと考えられる。そこで「1文字単位での出現確率が低い候補文字を組合せると、さらに出現確率が低下する」という特徴を利用することで、検索精度の向上に寄与しない候補文字を排除することができる。実際に図6のように“日本”に対しては、

（日本、目本、白本、曰本、臼本、  
 日木、目木、白木、曰木、  
 日不、目不、白不、

日天、目天、

日末)

の15通りの展開語で検索した場合と、すべてを組合せた25通りの展開語で検索した場合と比べて検索精度の劣化はほとんど生じない。その理由を以下の例で説明する。

#### 【0024】

上記“日”と“本”の各々の第一候補文字の出現確率を1/2、第二候補文字の出現確率を1/4、第三候補文字の出現確率を1/8、第四候補文字の出現確率を1/16、第五候補文字の出現確率を1/32、それ以降の候補文字の出現確率を1/32と仮定する。そして、各々の候補文字を組合せて累積した出現確率を算出すると、

“日本” 1/4、 “目本” 1/8、 “白本” 1/16、 “曰本” 1/32、 “臼本” 1/64、  
 “日木” 1/8、 “目木” 1/16、 “白木” 1/32、 “曰木” 1/64、 “臼木” 1/128、  
 “日不” 1/16、 “目不” 1/32、 “白不” 1/64、 “曰不” 1/128 “臼不” 1/256、  
 “日天” 1/32、 “目天” 1/64、 “白天” 1/128、 “曰天” 1/256、 “臼天” 1/512

“日末” 1/64、 “目末” 1/128、 “白末” 1/256、 “曰末” 1/512、 “臼末” 1/1,024

となる。このうち本実施例に示す通り左上半分の文字列を採用することにより、

$$1/4 + 1/8 \times 2 + 1/16 \times 3 + 1/32 \times 4 + 1/64 \times 5 = 57/64 \doteq 90\%$$

の確率で検索漏れを抑止することが可能となる。そのため、出現確率の小さな候補文字は対象から除外しても検索精度への影響はほとんどない。

#### 【0025】

(l、I、!、1、i) (o、O、0、Q、6) (c、C、G、e、q) (k、K、h、b、R)

また、“lo”の場合について図15を用いて説明すると、従来技術では各文字の候補文字が認識誤りし易い順に列挙された(l、I、!、1、i) (o、O、0、Q、6)に対して、すべてを組合せた

(lo、Io、!o、1o、io、  
 lO、IO、!O、1O、iO、

10、I0、!0、10、i0、

1Q、IQ、!Q、1Q、iQ、

16、I6、!6、16、i6)

の $5 \times 5 = 25$ 通りの展開語を生成し、それらの展開語の論理和集合を検索条件として検索を行っている。しかし、“i6”のような第5候補文字と第5候補文字の組合せの展開語が検索精度の向上に寄与する可能性は極めて低いと考えられる。そこで「1文字単位での出現確率が低い候補文字を組合せると、さらに出現確率が低下する」という特徴を利用することで、検索精度の向上に寄与しない候補文字を排除することができる。実際に図6のように“1o”に対しては、

(1o、I o、! o、1 o、i o、

1 O、I O、! O、1 O、

1 0、I 0、! 0、

1 Q、I Q、

1 6)

の15通りの展開語で検索した場合と、すべてを組合せた25通りの展開語で検索した場合と比べて検索精度の劣化はほとんど生じない。その理由を以下の例で説明する。

#### 【0026】

上記“1”と“o”の各々の第一候補文字の出現確率を $1/2$ 、第二候補文字の出現確率を $1/4$ 、第三候補文字の出現確率を $1/8$ 、第四候補文字の出現確率を $1/16$ 、第五候補文字の出現確率を $1/32$ 、それ以降の候補文字の出現確率を $1/32$ と仮定する。そして、各々の候補文字を組合せて累積した出現確率を算出すると、

“1o”  $1/4$ 、 “I o”  $1/8$ 、 “! o”  $1/16$ 、 “1 o”  $1/32$ 、 “i o”  $1/64$ 、

“1 O”  $1/8$ 、 “I O”  $1/16$ 、 “! O”  $1/32$ 、 “1 O”  $1/64$ 、 “i O”  $1/128$ 、

“1 0”  $1/16$ 、 “I 0”  $1/32$ 、 “! 0”  $1/64$ 、 “1 0”  $1/128$ 、 “i 0”  $1/256$ 、

“1 Q”  $1/32$ 、 “I Q”  $1/64$ 、 “! Q”  $1/128$ 、 “1 Q”  $1/256$ 、 “i Q”  $1/512$

“1 6”  $1/64$ 、 “I 6”  $1/128$ 、 “! 6”  $1/256$ 、 “1 6”  $1/512$ 、 “i 6”  $1/1,0$

となる。このうち本実施例に示す通り左上半分の文字列を採用することにより、  
 $1/4 + 1/8 \times 2 + 1/16 \times 3 + 1/32 \times 4 + 1/64 \times 5 = 57/64 \approx 90\%$   
 の確率で検索漏れを抑止することが可能となる。そのため、出現確率の小さな候補文字は対象から除外しても検索精度への影響はほとんどない。

## 【0027】

このようにして作成した類似文字テーブルの例を図16に示す。

## 【0028】

このようにn文字単位の文字列の出現確率を基に候補文字を選択することで、出現確率の高い候補文字に絞り込んだ類似文字テーブルとなり、検索精度向上に寄与する候補文字数を少なくすることが可能となる。以上、本類似文字テーブル109の説明である。

## 【0029】

以下、本文書検索システムにおける登録処理について図7を用いて説明する。

## 【0030】

文書の登録の際は、まず登録する紙文書をスキャナ103にセット（ステップ2000）し、キーボード101から入力されたコマンドを受け、システム制御プログラム201は文書登録制御プログラム202を起動する（ステップ2001）。この文書登録制御プログラム202は、最初にスキャナ制御プログラム203を起動して、スキャナ103にセットしてある紙文書からイメージデータを抽出し、ワークエリア216に出力する（ステップ2002）。次に文書登録制御プログラム202はOCR制御プログラム204を起動し、ワークエリア216のイメージデータを入力として文字認識を行い、テキストデータを抽出しワークエリア216に出力する（ステップ2003）。最後に文書登録制御プログラム202は文書登録プログラム205を起動し、ワークエリア216に読み込まれているテキストデータとイメージデータの識別子を関連付ける。テキストデータから検索用のインデクスデータを作成する。そして、テキストデータはテキストデータ106として、イメージデータは画像データ107として、磁気ディスク104へ格納する（ステップ2004）。なお、本実施例は紙文書をスキャナからイメージデータを入力するだけでなく、通信回線を介してFAXなどから直

接イメージデータを入力する構成をとってもかまわない。以上、本文書検索システムにおける登録処理の説明である。

【0031】

以下、本文書検索システムにおける検索処理について図8を用いて説明する。

【0032】

検索の際は、検索条件式がキーボード101から入力されると、システム制御プログラム201により展開制御プログラム206が起動される（ステップ2010）。次に展開制御プログラム206は最初に展開語生成プログラム207を起動して、入力された検索文字列に対して複数の展開語を生成しワークエリア216に出力する（ステップ2011）。次に展開制御プログラム206は検索条件式生成プログラム211を起動し、ワークエリア216に読み込まれている展開語の論理和（OR）集合となる検索条件式に拡張してシステム制御プログラム201に出力する（ステップ2012）。次にシステム制御プログラム201は検索制御プログラム212を起動し、出力された検索条件式を入力する。そして、本制御プログラムの下で検索条件解析プログラム213、テキストサーチプログラム214が順次起動され、検索条件式に従いテキストサーチを行う（ステップ2013）。最後に検索結果をシステム制御プログラム201に出力する（ステップ2014）。

【0033】

次に、展開語生成プログラム207の詳細な処理手順について図9を用いて説明する。展開語生成プログラム207は、検索用文字列分割プログラム208を起動し、入力された検索文字列を所定の $n$ 文字単位（ $n \geq 2$ ）の部分文字列に分割する（ステップ2020）。次に、類似文字テーブル参照プログラム209を実行して、分割された各部分文字列ごとの候補文字を上記で説明した $n$ 文字単位（ $n \geq 2$ ）の類似文字テーブル109より参照し、ワークエリア216に格納する（ステップ2021）。次に、検索文字列展開プログラム210を実行して、ワークエリア216から各部分文字列の候補文字を読み出して、それぞれを組合せることで複数の展開語を生成する（ステップ2022）。以上、本文書検索システムにおける展開語生成プログラム207の処理手順の説明である。

【0034】

以上、本文書検索システムにおける検索処理の説明である。

【0035】

以下、本文書検索システムにおける文書表示の処理について図1.0を用いて説明する。

【0036】

検索結果の中からユーザが指定した文書を表示する際は、ユーザが表示したい文書を指定する（ステップ2030）。すると、システム制御プログラム201が表示プログラム215を起動し、磁気ディスク104上のテキストデータ106を表示する（ステップ2031）。このとき、イメージデータでの表示を指定されたか否かを判定し（ステップ2032）、磁気ディスク104上の関連付けられたイメージデータ107を表示する（ステップ2033）。

【0037】

上記で説明した検索方法について、検索文字列として“日本文化”を用いた場合を例に具体的に説明する。この例では、2文字単位の展開とし“日本”と“文化”の候補文字を図6の類似文字テーブルから参照するものとする。

【0038】

検索文字列“日本文化”が入力されると、まず展開語生成の処理を行う。展開語生成では、まず検索文字列“日本文化”を2文字単位の部分文字列“日本”と“文化”に分割する。次に“日本”の候補文字を類似文字テーブルから参照し、

（日本、目本、白本、曰本、臼本、日木、目木、白木、曰木、日不、目不、白不、日天、目天、日末）

をワークエリアに読み込む。同様に“文化”の候補文字を

（文化、丈化、女化、夂化、大化、文仕、丈仕、女仕、夂仕、文牝、丈牝、女牝

文比、丈比、文北）

をワークエリアに読み込む。次に各部分文字列の候補文字を組合せることで、

“日本文化”

“日本文化”

“日本女化”

“日本文化”

“日本大化”

...

“日末文北”

の展開語を生成する。最後に生成した展開語のいずれかを含む文書を探す論理和 (OR) 条件「“日本文化” or “日本文化” or “日本女化” or “日本文化” or “日本大化” or … or “日末文北”」に従い検索を行うことで検索漏れを低減した検索が可能となる。このように長い検索文字列では、所定の長さの部分文字列単位で展開し、出現確率が低い候補文字を排除した類似文字テーブルを用いることで、従来の方法では  $5 \times 5 \times 5 \times 5 = 625$  通りの展開語による検索に対し、本発明では  $15 \times 15 = 225$  通りの展開語による検索となる。すなわち、出現確率の高い候補文字に絞った類似文字テーブルに基づいて展開される展開語の数は、従来技術のようにすべての候補文字の組合せから生成される展開語の数に比べて、検索精度を維持したまま大幅に削減することが可能である。このため大幅に検索時間を短縮することが可能となる。

#### 【0039】

さらに、上記で説明した検索方法について、検索文字列として“lock”を用いた場合を例に具体的に説明する。この例では、2文字単位の展開とし“lo”と“ck”の候補文字を図16の類似文字テーブルから参照するものとする。

#### 【0040】

検索文字列“lock”が入力されると、まず展開語生成の処理を行う。展開語生成では、まず検索文字列“lock”を2文字単位の部分文字列“lo”と“ck”に分割する。次に“lo”の候補文字を類似文字テーブルから参照し、(lo、Io、!o、lo、io、lO、IO、!O、lO、lO、IO、!O、lQ、IQ、l6) をワークエリアに読み込む。同様に“ck”の候補文字 (ck、Ck、Gk、ek、qk、cK、CK、GK、eK、ch、Ch、GH、cb、Cb、cR)

をワークエリアに読み込む。次に各部分文字列の候補文字を組合せることで、

“l o c k”

“l o C k”

“l o G k”

“l o e k”

“l o q k”

...

“l 6 c R”

の展開語を生成する。最後に生成した展開語のいずれかを含む文書を探す論理和 (OR) 条件「“l o c k” or “l o C k” or “l o G k” or “l o e k” or “l o q k” or ... or “l 6 c R”」に従い検索を行うことで検索漏れを低減した検索が可能となる。このように長い検索文字列では、所定の長さの部分文字列単位で展開し、出現確率が低い候補文字を排除した類似文字テーブルを用いることで、従来の方法では  $5 \times 5 \times 5 \times 5 = 625$  通りの展開語による検索に対し、本発明では  $15 \times 15 = 225$  通りの展開語による検索となる。すなわち、出現確率の高い候補文字に絞った類似文字テーブルに基づいて展開される展開語の数は、従来技術のようにすべての候補文字の組合せから生成される展開語の数に比べて、検索精度を維持したまま大幅に削減することが可能である。このため大幅に検索時間を短縮することが可能となる。

#### 【0041】

以上、第一の実施例を説明した。本実施例によれば、OCRによる認識誤りを許容した検索において、検索漏れの発生を低減し、高い検索精度の検索を実用的な検索時間で可能となる。

#### 【0042】

次に、本発明の第二の実施例について説明する。

#### 【0043】

第一の実施例では、n文字単位の類似文字テーブルを参照することにより、検索精度に寄与する確率の低い文字列を展開の対象から除外する。これにより検索文字列が長い場合にも高速な検索を実現している。しかし、本方法では文書検索

時に指定した検索文字列が短い場合、認識誤りする可能性のある語に展開して検索することで所望しない結果（以下、検索ノイズとする）が増えてしまう。例えば、検索文字列が“犬”の場合、この検索文字列を展開することで「“犬” or “尤” or “大” or “太” …」の論理和集合で検索すると、“大”や“太”などのような別の意味を持つ展開語を含む文書もまた検索結果となる。そのため、検索ノイズが多くなり検索精度が劣化する。

## 【 0 0 4 4 】

第二の実施例では、第一の実施例に加えて、入力された検索文字列の長さで展開するか否かを判定し、展開方法を切り替えるステップを備えることで、検索文字列が短い場合では検索ノイズを低減するという効果が得られる。

## 【 0 0 4 5 】

図 1 1 は、本実施例を説明する構成図である。本実施例と第一の実施例は基本的には同様であるが、異なる点は展開制御プログラム 2 0 6 に新たに展開方法切り替えプログラム 3 0 0 が追加された構成となる。

## 【 0 0 4 6 】

文書の登録方法は第一の実施例と同様であるので省略し、検索方法について図 1 2 を用いて説明する。

## 【 0 0 4 7 】

検索の際は、検索条件式がキーボード 1 0 1 から入力されると、システム制御プログラム 2 0 1 により展開制御プログラム 2 0 6 が起動される（ステップ 3 0 0 0）。次に展開制御プログラム 2 0 6 は最初に展開方法を切り替えプログラム 3 0 0 を起動して、入力された検索文字列の長さを取得する（ステップ 3 0 0 1）。そして、取得した長さを判定（ステップ 3 0 0 2）し、所定の長さを超えなければ展開しないものとして入力された検索条件式を維持したまま（ステップ 3 0 0 5）に進み、所定の長さを超えれば（ステップ 3 0 0 3）に進む。展開制御プログラム 2 0 6 は展開語生成プログラム 2 0 7 を起動して、入力された検索文字列に対して複数の展開語を生成しワークエリア 2 1 6 に出力する（ステップ 3 0 0 3）。次に展開制御プログラム 2 0 6 は検索条件式生成プログラム 2 1 1 を起動し、ワークエリア 2 1 6 に読み込まれている展開語の論理和（OR）集合と

なる検索条件式に拡張してシステム制御プログラム 2 0 1 に出力する（ステップ 3 0 0 4）。次にシステム制御プログラム 2 0 1 は検索制御プログラム 2 1 3 を起動し、元の検索条件式または出力された検索条件式を入力する。そして、本制御プログラムの下で検索条件解析プログラム 2 1 2、テキストサーチプログラム 2 1 4 が順次起動し、検索条件式に従いテキストサーチを行う（ステップ 3 0 0 5）。最後に検索結果をシステム制御プログラム 2 0 1 に出力する（ステップ 3 0 0 6）。以上、本文書検索システムにおける検索処理の説明である。

## 【 0 0 4 8 】

上記で説明した検索方法について、検索文字列として“犬”を用いた場合を例に具体的に説明する。この例では、検索文字列の展開判定の所定値を 1 とする。

## 【 0 0 4 9 】

検索文字列“犬”が入力されると、まず検索文字列の長さ 1 が取得される。次に展開判定において、取得した検索文字列の長さが所定値以下なので、展開語生成の処理を行わない。そのため、入力された検索文字列“犬”による検索を行う。このように短い検索文字列では、展開しないことで従来の技術のように別の意味の文字列を含む文書が結果とならないので検索ノイズを減らすことが可能となる。

## 【 0 0 5 0 】

また、本実施例では展開判定の所定値を予め設定するだけではなく、検索時に自由に調整することが可能である。さらに、漢字のような表意文字は 1 文字、英字などの表音文字は 2 文字のように文字種で切り替える構成も可能である。

## 【 0 0 5 1 】

以上、第二の実施例を説明した。本実施例によれば、OCR による認識誤りを許容した検索において、検索文字列長が短い場合には検索ノイズが増加しない高精度な検索が可能となる。

## 【 0 0 5 2 】

次に、本発明の第三の実施例について説明する。

## 【 0 0 5 3 】

第三の実施例では、第一の実施例に加えてさらに類似文字テーブルの見出し文

字を全文字コードの組合せから一部分を抽出して作成することにより、類似文字テーブルのファイル容量を低減できるという効果がある。

## 【 0 0 5 4 】

すなわち、第一の実施例では、類似文字テーブルの見出し文字を全文字コードの組合せた学習データから作成している。この場合、日本語の全文字コードを約 8, 0 0 0 種とし、1 個の見出し文字について 1 0 個の候補文字を格納するケースを想定すると 2 文字単位類似文字テーブルの容量は以下の通りになる。

## 【 0 0 5 5 】

(全文字コードの組合せ) × 4 [バイト] (2 文字なので) × 1 0 個 =  
8, 0 0 0 × 8, 0 0 0 × 4 × 1 0 = 2. 5 6 G [バイト]

第三の実施例では、検索文字列として使用される確率の高い主要な文字列のみを 2 文字単位類似文字テーブルに格納することにより、類似文字テーブルの少容量化を実現しようとするものである。

## 【 0 0 5 6 】

本実施例と第一の実施例は基本的には同様であるが、異なる点は、類似文字テーブル 1 0 9 において、第一の実施例では  $n$  文字単位 ( $n \geq 2$ ) の候補文字を全文字コードの組合せで作成していたが、本実施例では検索文字列に使われる主要な文字の組合せに対してのみ作成している。そのため、類似文字テーブルにない見出し文字が存在するので、例外処理が類似文字テーブル参照プログラム 2 0 9 に追加されている。なお、本実施例の主要な組合せとしては、第一水準文字の組合せによるものを想定している。

## 【 0 0 5 7 】

以下、本実施例の類似文字テーブルを用いた際の展開処理の手順、すなわち展開語生成プログラム 2 0 7 の新たな処理手順について図 1 3 を用いて説明する。

## 【 0 0 5 8 】

展開語生成プログラム 2 0 7 では、検索用文字列分割プログラム 2 0 8 を起動し、入力された検索文字列を所定の  $n$  文字単位 ( $n \geq 2$ ) の部分文字列に分割する (ステップ 3 0 0 0)。次に、類似文字テーブル参照プログラム 2 0 9 を実行して、まず対象となる部分文字列が類似文字テーブル 1 0 9 の見出し文字に有る

か否か走査する（ステップ 3 0 0 1）。見出し文字があるか判定（ステップ 3 0 0 2）し、見出し文字がある場合は、候補文字を参照しワークエリア 2 1 6 に格納する（ステップ 3 0 0 3）。見出し文字がない場合は、部分文字列そのものをワークエリア 2 1 6 に格納する（ステップ 3 0 0 4）。最後に、検索文字列展開プログラム 2 1 0 を実行して、ワークエリア 2 1 6 から各部分文字列の候補文字または部分文字列を読み出して、それぞれを組合せることで複数の展開語を生成する（ステップ 3 0 0 5）。以上、本文書検索システムにおける展開処理の手順の説明である。

## 【 0 0 5 9 】

次に本実施例で用いている主要な組合せ文字における類似文字テーブルのファイル容量について示す。第一水準文字の数を約 3, 0 0 0 種とし、1 個の見出し文字について 1 0 個の候補文字を保持させると、類似文字テーブルのファイル容量は、

（第一水準文字の組合せ）× 4 [バイト]（2 文字なので）× 1 0 個 =  
 $3, 0 0 0 \times 3, 0 0 0 \times 4 \times 1 0 = 3 6 0 \text{ M [バイト]}$  の容量となる。すなわち、実施例 1 と比較して類似文字テーブルのファイル容量が約 1 / 7 で済むことになる。

## 【 0 0 6 0 】

また、本実施例では第一水準文字の組合せによるものだけではなく、新聞記事や各種文献などのコーパスに存在する文字の組合せを抽出し、言語として接続する文字の組合せをさらに絞り込むことが可能である。

## 【 0 0 6 1 】

以上、第三の実施例を説明した。本実施例によれば、OCR による認識誤りを許容した検索文字列で用いる類似文字テーブルの見出し文字において検索に使われる主要な文字の組合せに絞り込むことで、類似文字テーブルのファイル容量を大幅に削減することが可能となる。

## 【 0 0 6 2 】

なお、第三の実施例では、検索文字列に対して n 文字単位の類似文字テーブルを参照する際に、n 文字単位類似文字テーブルに記載されていない文字列につい

ては候補文字列展開の対象として組入れない方法について記載をしている。しかし、主要な文字列を対象として作成した  $n$  文字単位の類似文字テーブルと併用する形で  $m$  文字単位 ( $m < n$ ) の類字文字テーブルを予め作成しておき、 $n$  文字単位の類似文字テーブルに記載されていない文字列については、 $m$  文字単位の類似文字テーブルを参照することにより展開語を生成する構成を採ることも可能である。

#### 【0063】

次に、本発明の第四の実施例について説明する。

#### 【0064】

第一の実施例から第三の実施例では展開処理と検索処理を独立とする構成であったが、第四の実施例では展開処理を検索処理の中に組込んだ構成に拡張したものである。図14は、本実施例を説明する構成図である。これまでの実施例と異なり、検索の際には検索制御プログラム212で展開処理も含めて制御する。また、検索処理内部で検索文字列の展開を行っているため、新たに検索条件式を生成する検索条件式生成プログラム211を必要としない。

#### 【0065】

##### 【発明の効果】

以上のように本発明によれば、イメージ文書をOCRで文字認識した際に発生する認識誤りを含んだテキストデータを対象とした検索において、出現確率の低い  $n$  文字単位の候補文字を排除した類似文字テーブルから生成される展開語による検索を行い展開語数を低減することで、高い検索精度でありながら実用的な検索時間での検索を実現することが可能となる。

##### 【図面の簡単な説明】

##### 【図1】

第一の実施例の文書検索システムの構成図である。

##### 【図2】

従来技術による検索方法のフローチャートである。

##### 【図3】

本発明による検索方法のフローチャートである。

【図 4】

第一の実施例における類似文字テーブル作成の概要図である。

【図 5】

第一の実施例における類似文字テーブル作成の処理手順のフローチャートである。

【図 6】

第一の実施例における類似文字テーブルの一例である。

【図 7】

第一の実施例における文書登録の処理手順のフローチャートである。

【図 8】

第一の実施例における検索の処理手順のフローチャートである。

【図 9】

第一の実施例における展開語生成の処理手順のフローチャートである。

【図 1 0】

第一の実施例における文書表示の処理手順のフローチャートである。

【図 1 1】

第二の実施例の文書検索システムの構成図である。

【図 1 2】

第二の実施例における検索の処理手順のフローチャートである。

【図 1 3】

第三の実施例における展開語生成制御の処理手順のフローチャートである。

【図 1 4】

第四の実施例の文書検索システムの構成図である。

【図 1 5】

第一の実施例における類似文字テーブル作成の概要図である。

【図 1 6】

第一の実施例における類似文字テーブルの一例である。

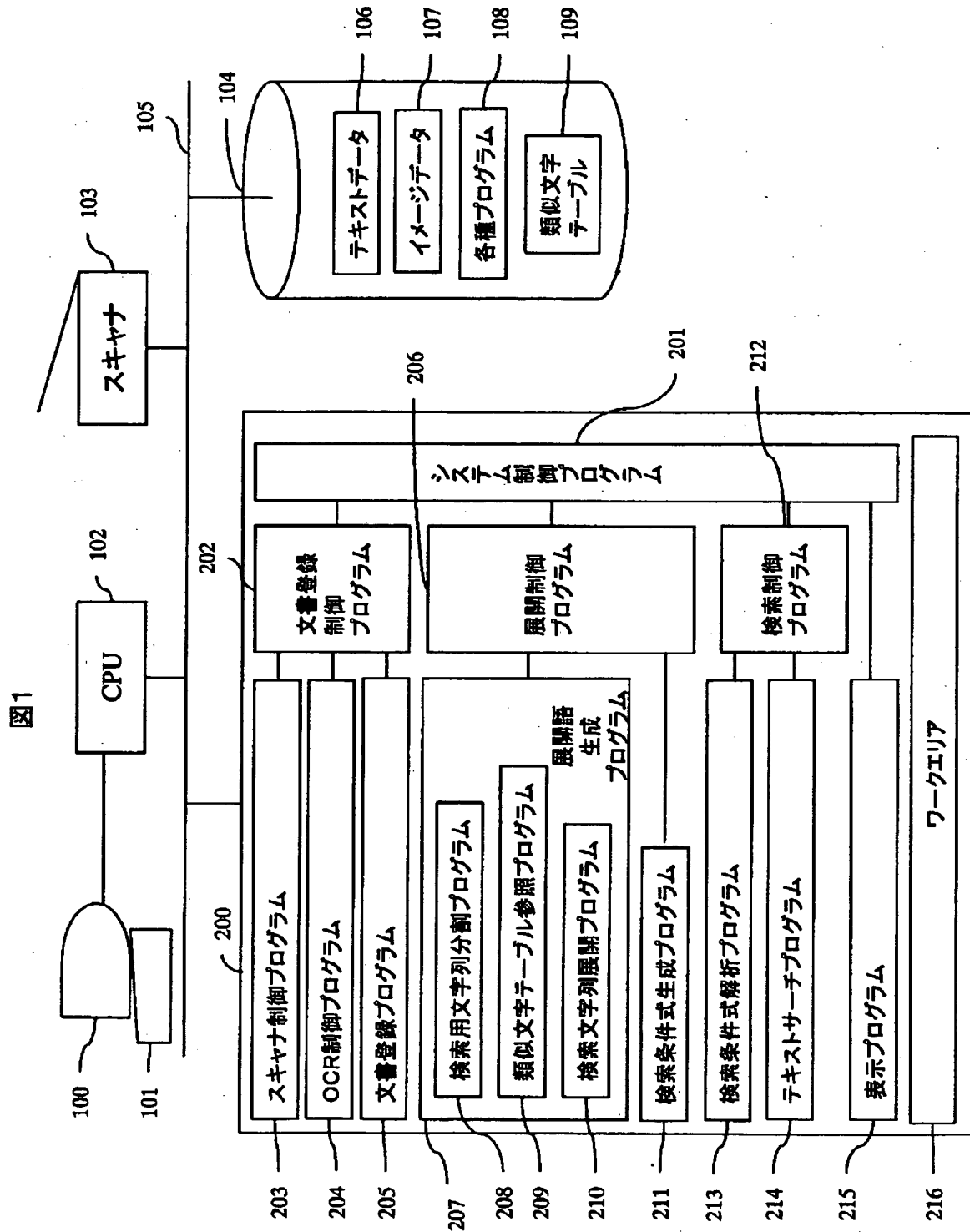
【符号の説明】

1 0 0 …ディスプレイ、1 0 1 …キーボード、1 0 2 …中央演算装置 CPU、1

03…スキャナ、104…磁気ディスク、105…バス、108…各種プログラム、109…類似文字テーブル、200…主メモリ。

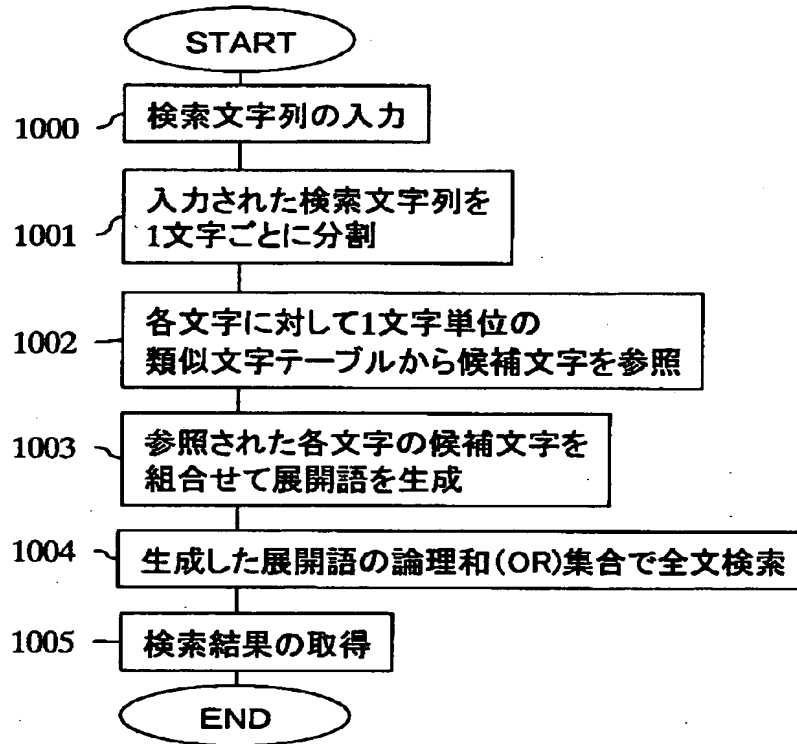
【書類名】 図面

【図 1】



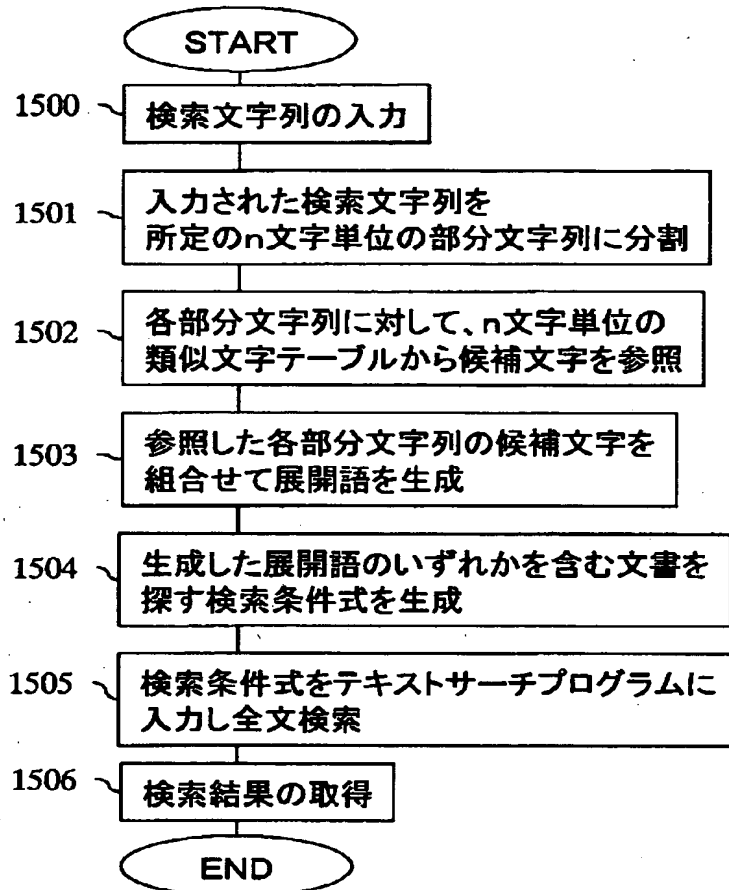
【図 2】

図2



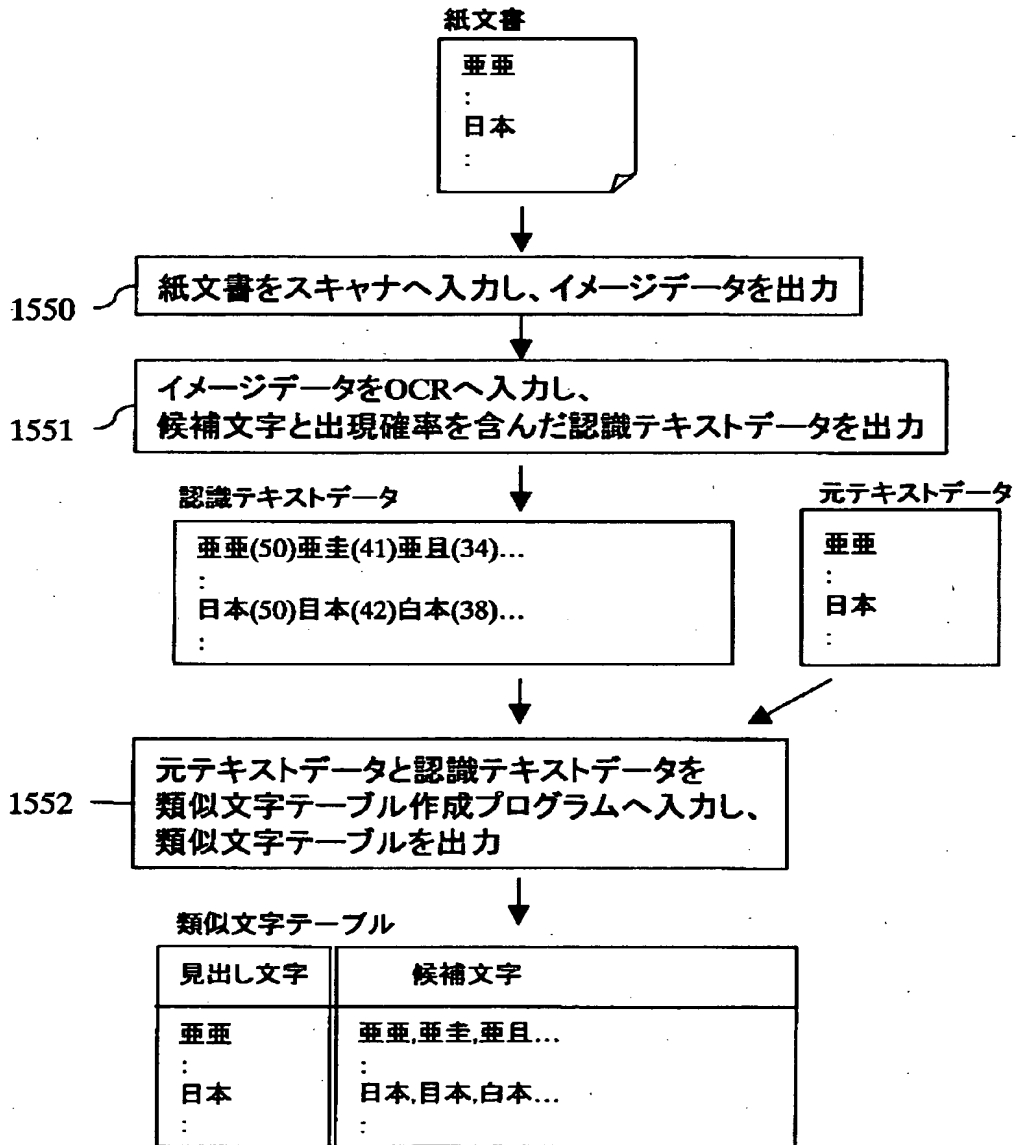
【図 3】

図3



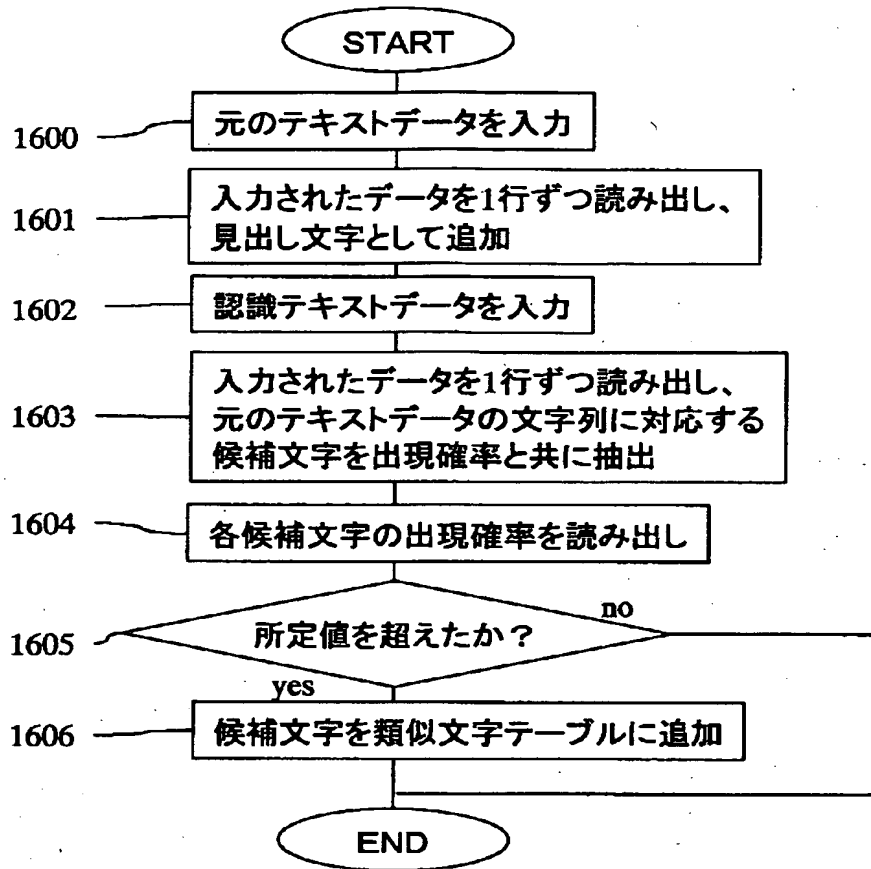
【図 4】

図4



【図 5】

図5



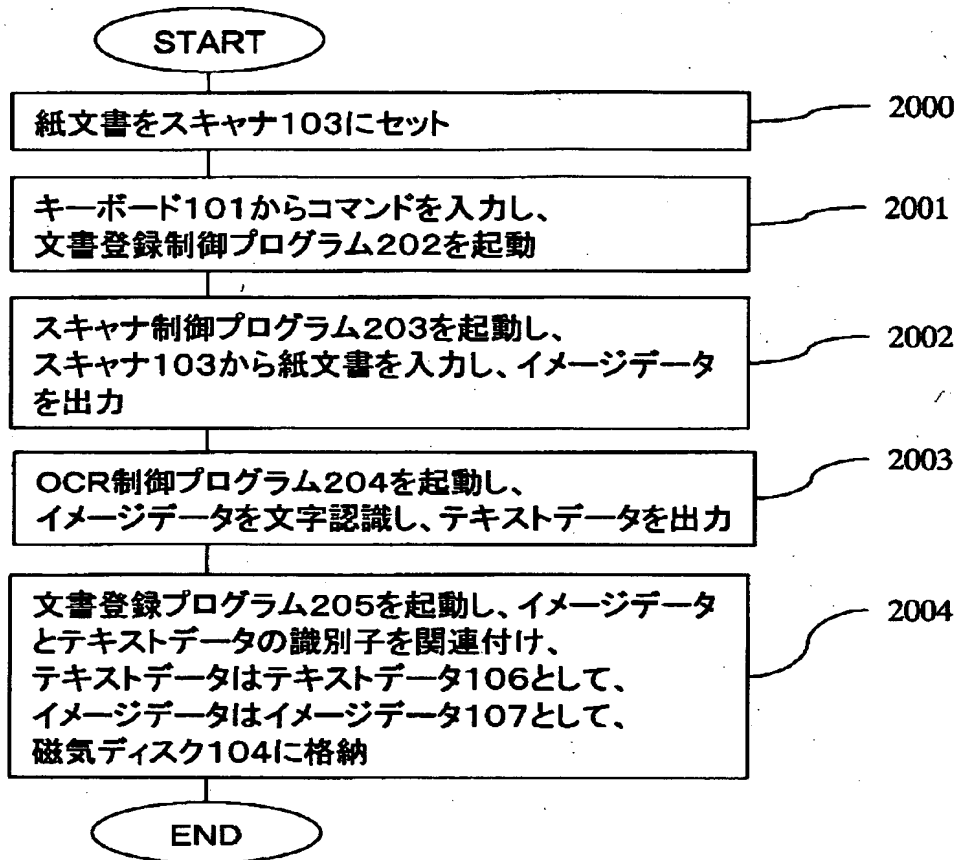
【図 6】

図6

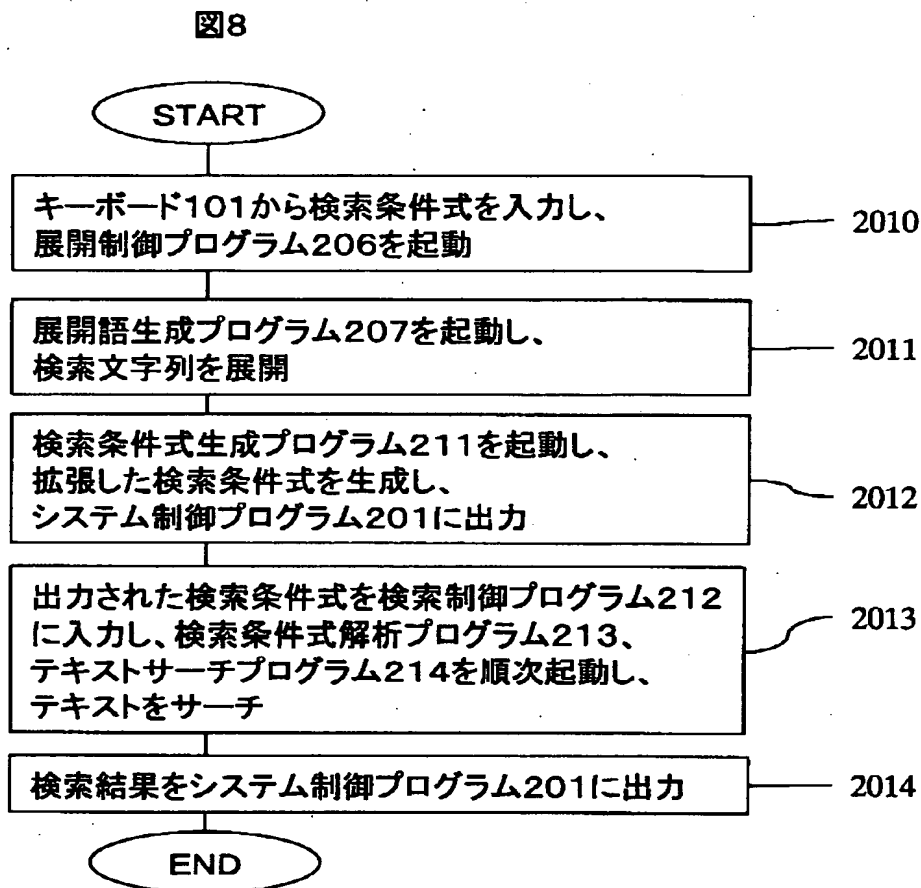
見出し文字	候補文字
日本 : 文化 :	日本,日本,白本,日本,白本,日木,目木,白木,日木,日不,目不,白不,日天,目天,日末 : 文化,文化,女化,文化,大化,文仕,文仕,女仕,女仕,文牝,文牝,女牝,文比,文比,文北 :

【図 7】

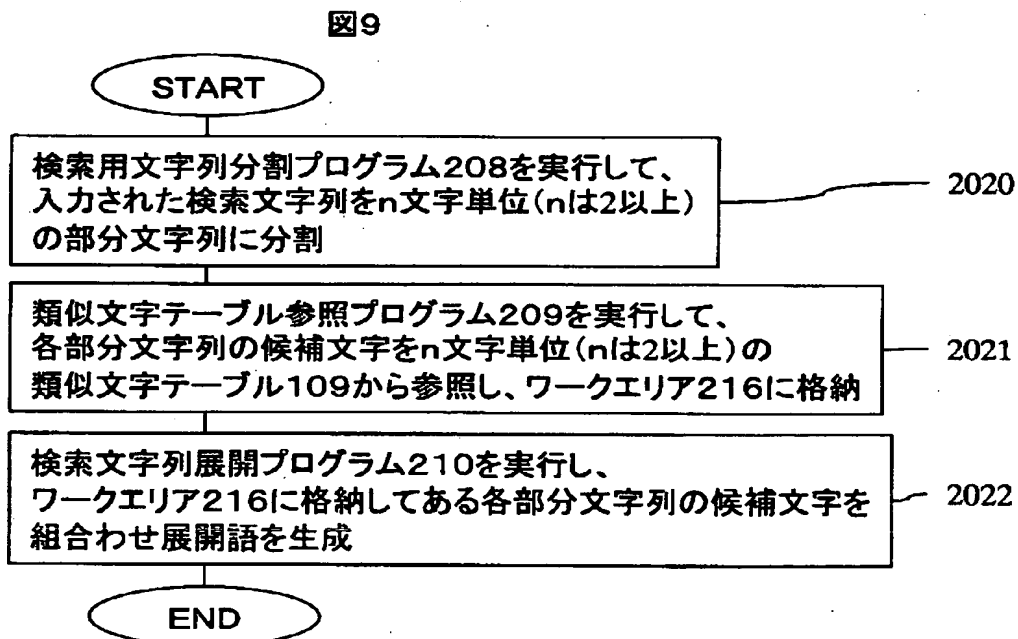
図7



【図 8】

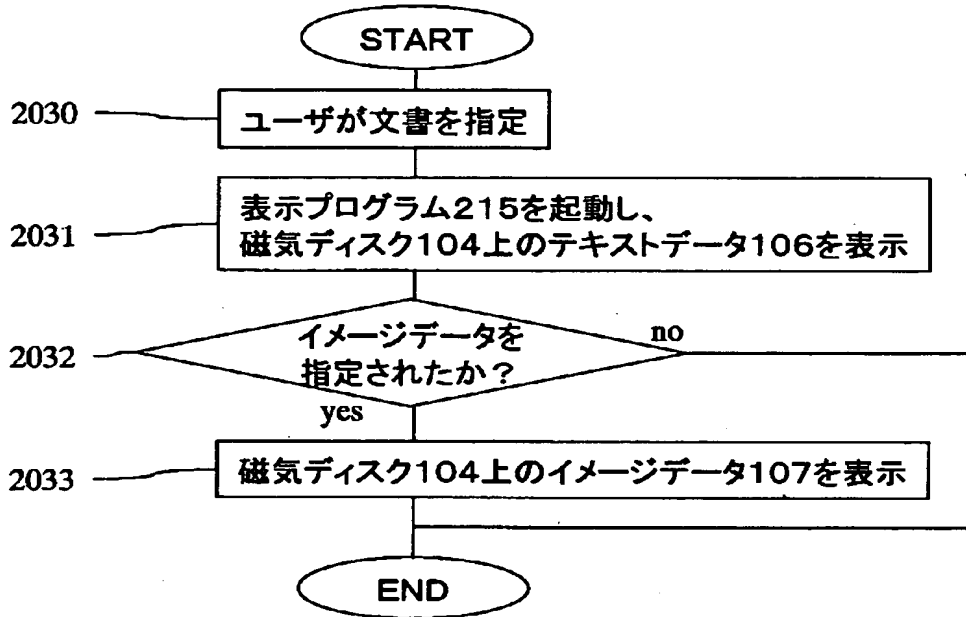


【図 9】

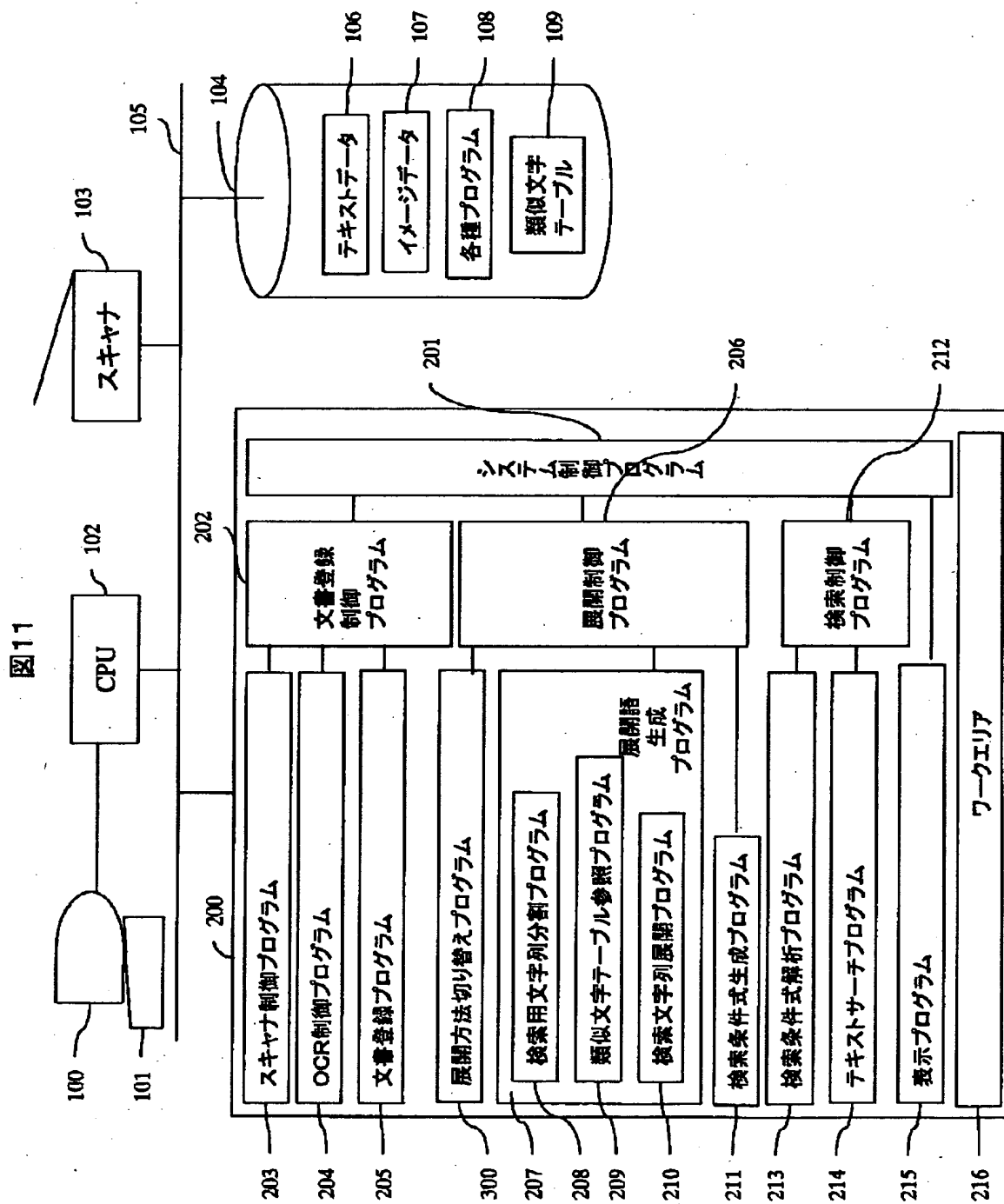


【図10】

図10

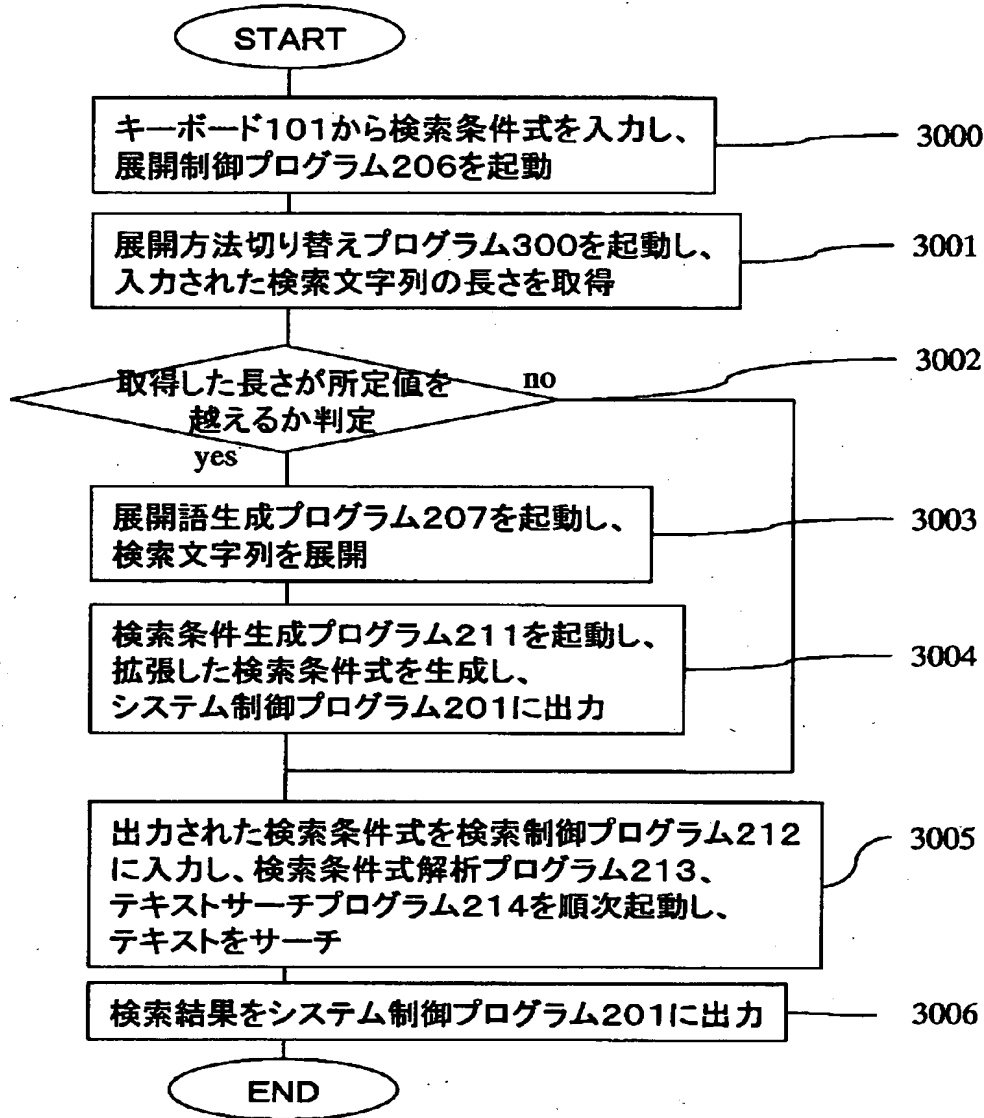


【図 11】



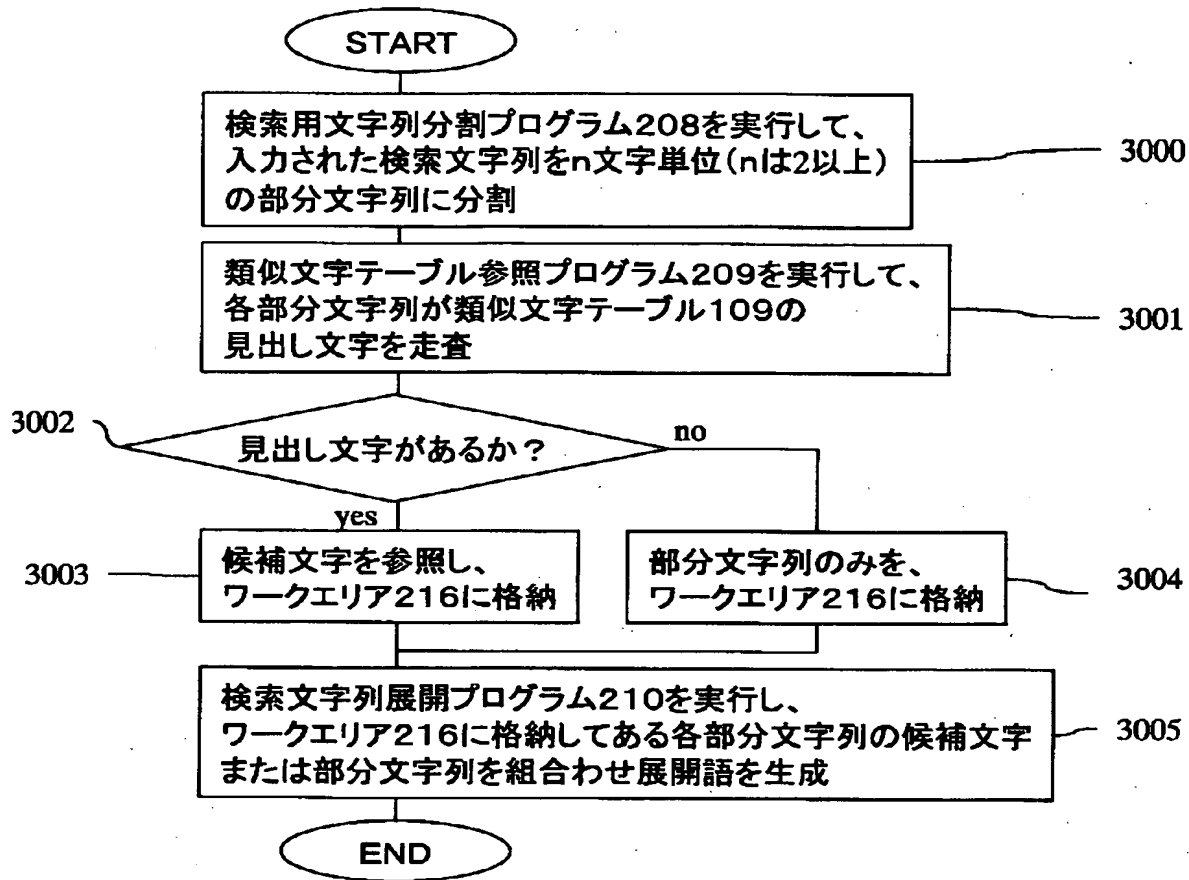
【図 1 2】

図12

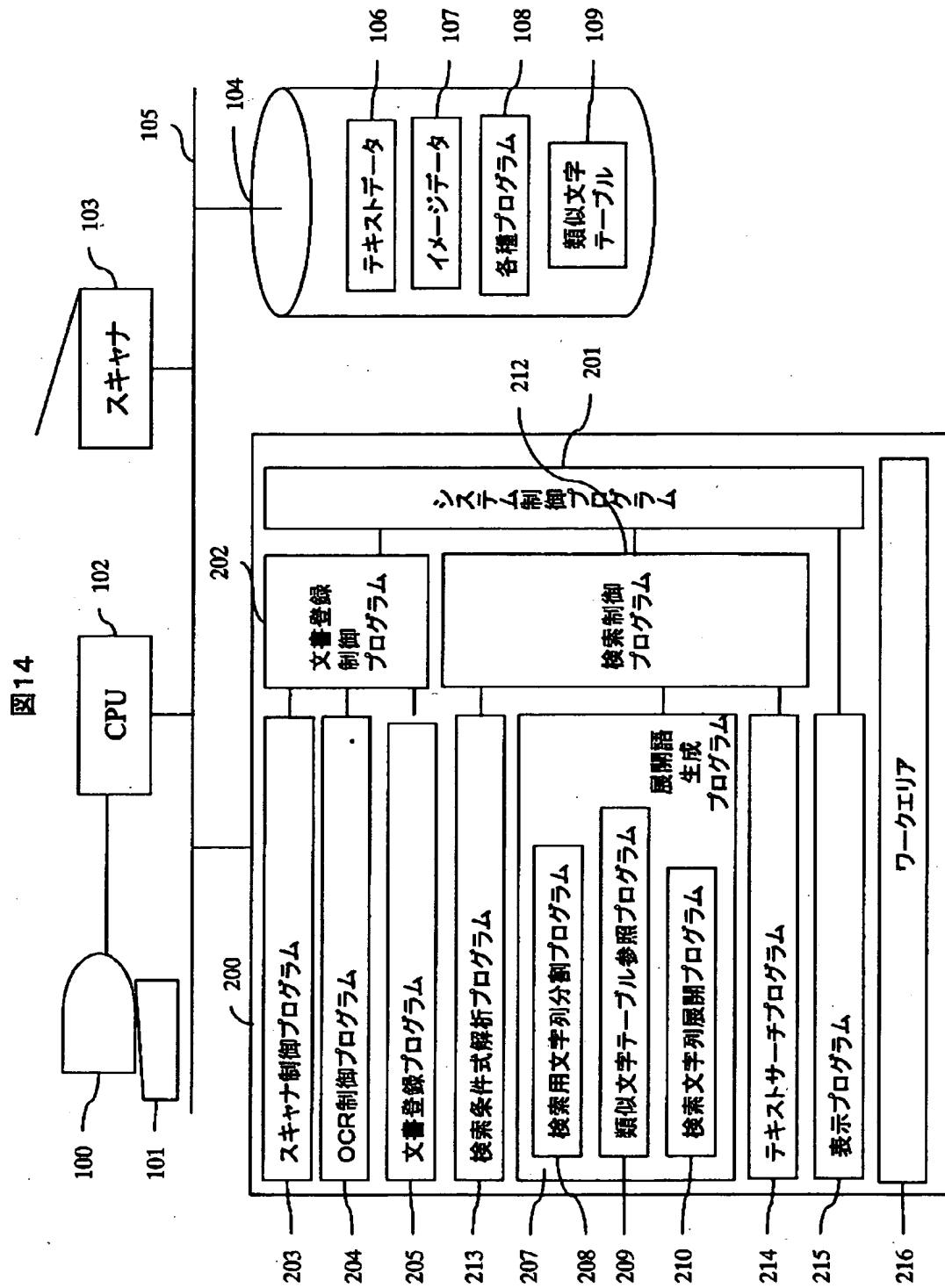


【図 1 3】

図13

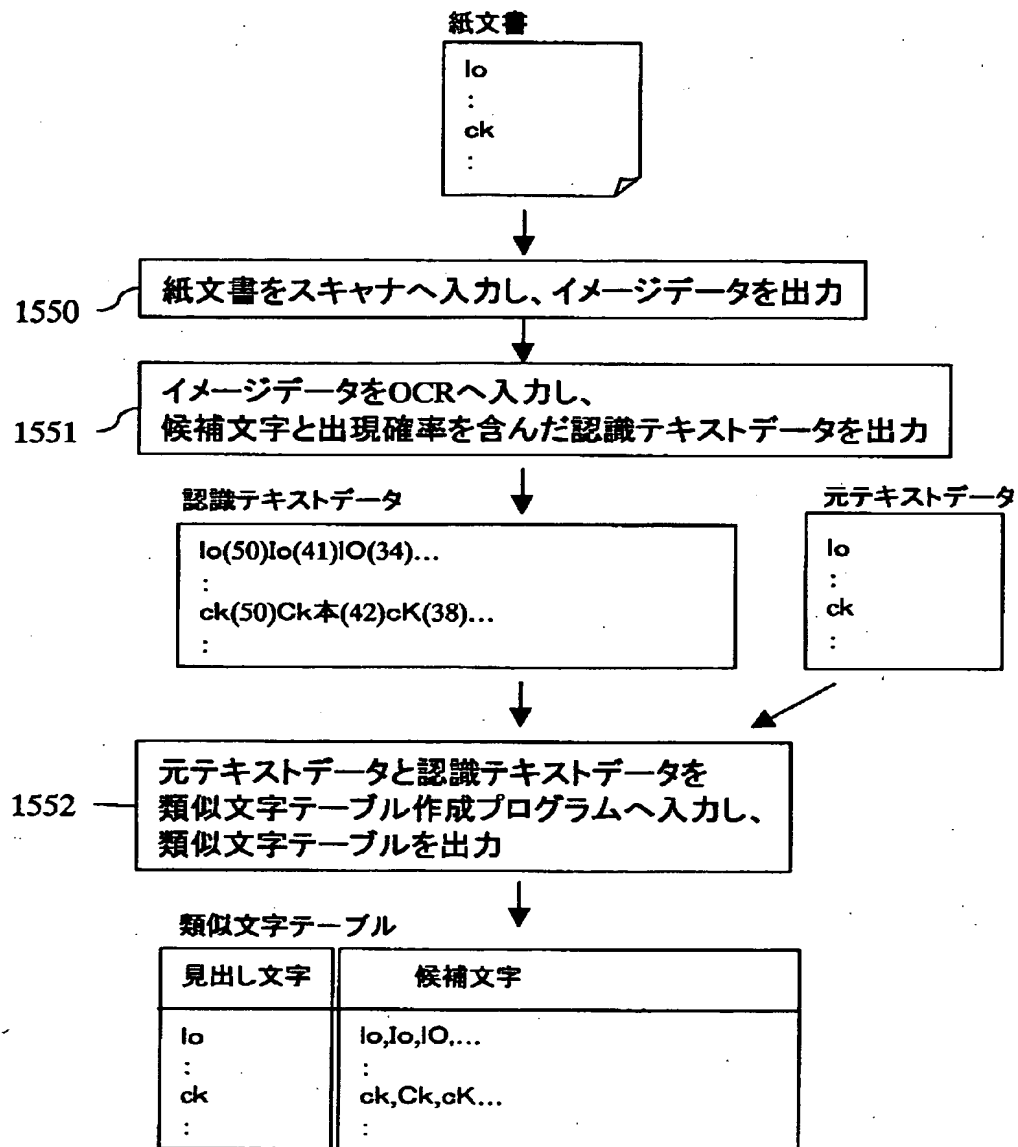


【図 14】



【図 1 5】

図15



【図 1 6】

図16

500	類似文字テーブル	501
見出し文字	候補文字	
lo	lo,lo, ! o, 1o,io,IO,IO, ! O,1O,I0,I0, ! 0,IQ,IQ,I6	
:	:	
ck	ck,Ck,Gk,ek,qk,cK,CK,GK,eK,ch,Ch,GH,cb,Cb,cR	
:	:	

【書類名】 要約書

【要約】

【課題】

OCRによる認識誤りを含む文書データベースを対象として、OCRの認識誤りによる検索漏れを抑止した高精度な検索を、比較的文字列長が長い検索タームが入力された場合にも実用的な時間で実現することを可能とする。

【解決手段】

イメージ文書を対象とした文字認識処理を実行した結果出力されるテキストによる文書を対象として、検索者が指定した検索文字列を含む文書を検索するシステムにおいて、前記検索文字列を所定の $n$ 文字単位の部分文字列( $n \geq 2$ )に分割する検索用文字列分割ステップと、前記 $n$ 文字単位の部分文字列( $n \geq 2$ )に対して、誤認識される可能性の高い文字形状の類似した類似文字列を格納することにより予め作成した $n$ 文字単位の類似文字テーブルを参照する類似文字テーブル参照ステップと、前記検索文字列を構成する部分文字列に対して $n$ 文字単位類似文字テーブルを参照することにより抽出し類似文字列群を組合せて展開語を生成する検索文字列展開ステップを有する。

【選択図】 図1

出 願 人 履 歴 情 報

識別番号 [000005108]

1. 変更年月日	1990年 8月31日
[変更理由]	新規登録
住 所	東京都千代田区神田駿河台4丁目6番地
氏 名	株式会社日立製作所

出 願 人 履 歴 情 報

識別番号 [391002409]

1. 変更年月日 2000年 3月30日

[変更理由] 名称変更

住 所 東京都大田区大森北3丁目2番16号

氏 名 株式会社 日立システムアンドサービス